2.41 MISSILE MOVEMENT

The importance of missile movement is its direct relationship to whether or not the missile will intercept the target and its degree of success or failure in the destruction of the target. The movement of any physical object is caused by the application of forces. The application of a force to a body causes it to accelerate proportionally to its mass. This is an instantaneous and continuous process in a physical system. To simulate how the application of a force accelerates and moves an object, the continuous nature of the process must be divided into discrete time steps. If the time steps are small, the acceleration can be assumed to be unchanging over the course of the time step which simplifies the integration process.

If we have an accurate representation of the forces and moments acting on a missile and the mass and moments of inertia of the missile, we can then construct the missile's trajectory and orientation in inertial space. The forces acting on the missile are aerodynamic (lift and drag), engine thrust and gravity. The moments are due to aerodynamic forces which cause the missile to rotate.

The forces and moments are computed in the body axis coordinate system but we require the missile trajectory in inertial space. This is accomplished by relating the body rotations to inertial space using a direction cosine transformation matrix (developed in Force and Moment Generation FE). Applying this matrix to the body directional forces, the components of inertial accelerations are determined. These in turn are used to update the missile's inertial location, orientation, velocity and angular rotation rates. The missile's orientation is defined by what are known as Euler angles where θ is pitch, ψ is yaw and φ is roll.

To update the "position" elements of the missile (location and orientation) the velocity and accelerations of the missile are required. For location, linear velocity and acceleration and for orientation, angular velocity and acceleration. Updating the "velocity" elements of the missile (linear velocity and angular velocity) requires the linear or angular accelerations.

2.41.1 Functional Element Design Requirements

This section contains the design requirements necessary to fully implement the missile movement simulation.

a. ESAMS will calculate the inertial accelerations caused by the application of body forces.

b. ESAMS will compute the angular accelerations around the body axes caused by the application of moments.

c. ESAMS will provide updated values for the missile's position, velocity, orientation and angular rotations for each time step.

2.41.2 Functional Element Design Approach

This section describes the design approach (equations and algorithms) implementing the design requirements of the previous section.

ESAMS uses a standardized body axis system centered on the center of gravity defined as follows:

a. X_B-axis, called the roll axis, longitudinally along the body of the missile positive forward.

b. Y_B-axis, called the pitch axis, laterally sidewards, positive to the left if viewing the missile from the rear.

c. Z_B-axis, called the yaw axis, laterally upward, positive up to form a right handed system with the other two.

A three-dimensional system has 6 degrees of freedom, 3 translational and 3 angular. Table 2.41-1 defines terms for the forces and moments acting on the missile, the linear and angular velocities, the moments of inertia and the inertial accelerations. These quantities are shown in Figures 2.41-1 and 2.41-2. Figure 2.41-2 also shows the definitions of the inertial coordinate system into which the body forces must be translated.

TABLE 2.41-1. Missile Movement Symbol Definitions.

	Roll Axis X_B	**Pitch Axis Y_B**	**Yaw Axis Z_B**
Angular rates	p	q	r
Components of missile inertial velocity	V_{Ix}	V_{Iy}	V_{Iz}
Components of force acting on missile along each axis	F_x	F_y	F_z
Moments acting on missile about each axis	M_x	M_y	M_z
Moments of inertia about each axis	A	B	C
Components of inertial accelerations	a_{Ix}	a_{Iy}	a_{Iz}
Euler Angles	φ	θ	ψ

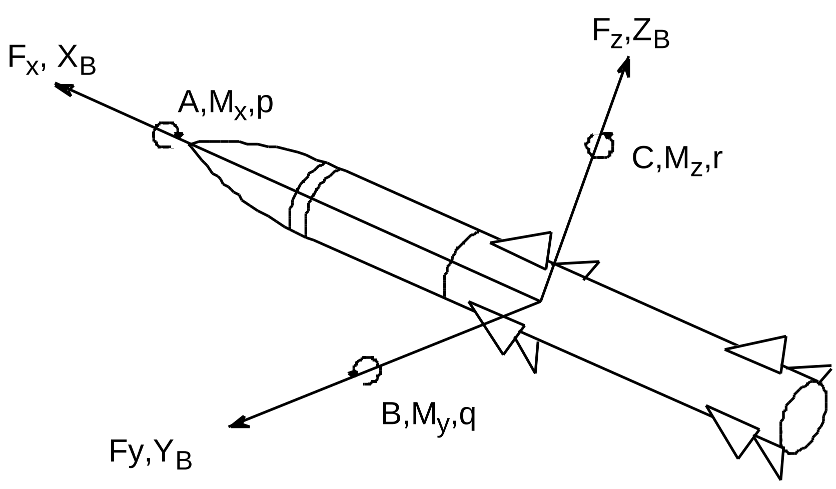

FIGURE 2.41-1. Missile Movement Body Axis Conventions.

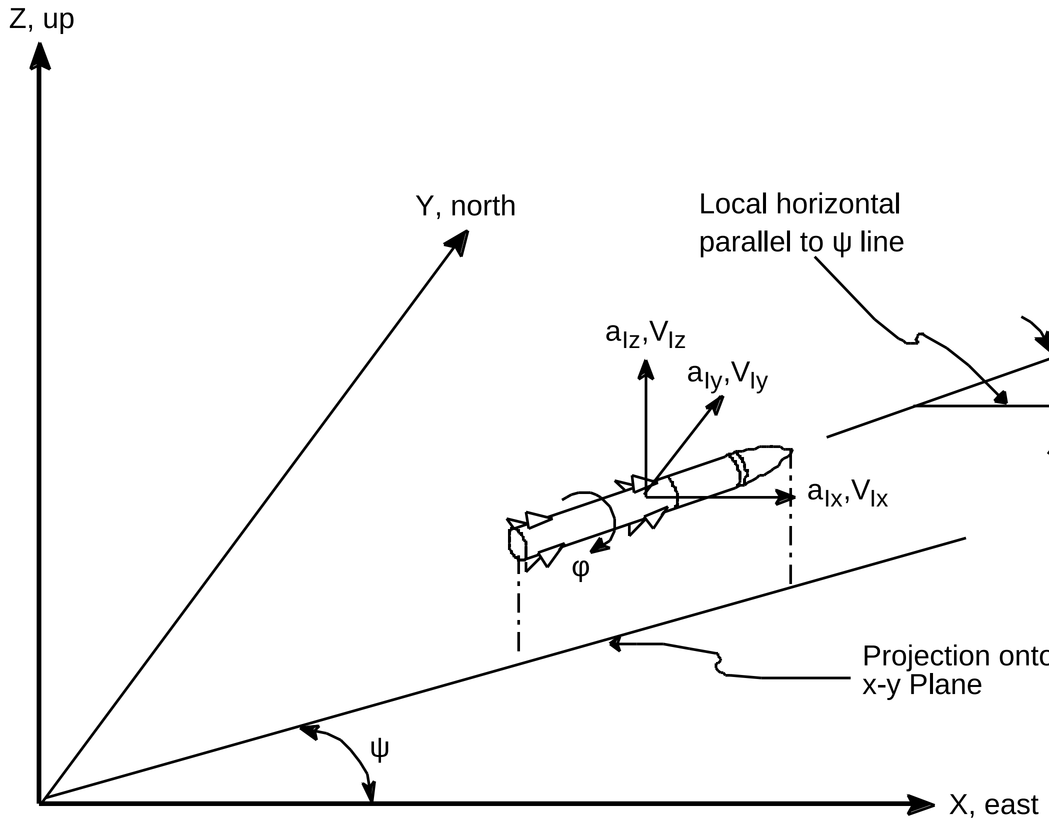

FIGURE 2.41-2. Missile Movement Inertial Conventions.

Design Element 41-1: Compute Inertial Accelerations

$$\vec{F} = m\vec{a} \qquad [2.41\text{-}1]$$

Newton's second law states that a force acting on a body is equal to its mass times the induced acceleration. This equation is directional, however. A force acting in a particular direction will cause an acceleration in that direction. In our case, the aerodynamic forces we have already determined, F_x, F_y and F_z, (see Force and Moment Generation) act in the body directions. The forces derived in force and moment generation were those caused by aerodynamic forces acting on the body due to lift and drag. Two additional forces are also acting on the missile body, engine thrust and gravity. Thrust is a body force so it can be added directly to the forces in the F_x direction, as shown below. The inertial force due to gravity will be accounted for when the inertial accelerations have been computed.

$$F_x = T - D$$
$$F_y = L_y \qquad [2.41\text{-}2a,b,c]$$
$$F_z = L_z$$

where: T = Thrust
D = Drag force
L_y = Lift in y-body direction
L_z = Lift in z-body direction

To derive the inertial components of acceleration that these body forces will produce, a relationship between the two coordinate systems is needed. This relationship was derived and explained in the force and moment generation functional element and was called the direction cosine transformation matrix. At that time, the direction cosine transformation matrix was derived to transform inertial rotations into the body system. Now we require body system components transformed to inertial components. To do this, the direction cosine transformation matrix is transposed. Applying the transposed matrix in the development of inertial accelerations yields the following matrix equation:

$$\begin{bmatrix} a_{Ix} \\ a_{Iy} \\ a_{Iz} \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & -\sin\varphi\sin\theta\cos\psi - \cos\varphi\sin\psi & -\cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi \\ \cos\theta\sin\psi & -\sin\varphi\sin\theta\sin\psi + \cos\varphi\cos\psi & -\cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi \\ \sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{bmatrix} \begin{bmatrix} F_x / m \\ F_y / m \\ F_z / m \end{bmatrix} \qquad [2.41\text{-}3]$$

Multiplying this equation out, substituting the following terms for the trigonometric relations and accounting for gravity gives:

$\rho_x = cos\theta cos\psi$
$\rho_y = cos\theta sin\psi$
$\rho_z = sin\theta$
$\pi_x = -sin\varphi sin\theta cos\psi - cos\varphi sin\psi$
$\pi_y = -sin\varphi sin\theta sin\psi + cos\varphi cos\psi$
$\pi_z = sin\varphi cos\theta$
$\eta_x = -cos\varphi sin\theta cos\psi + sin\varphi sin\psi$
$\eta_y = -cos\varphi sin\theta sin\psi - sin\varphi cos\psi$
$\eta_z = cos\varphi cos\theta$

$$a_{Ix} = \frac{\left(\rho_x F_x + \pi_x F_y + \eta_x F_z\right)}{m} \qquad [2.41\text{-}4]$$

$$a_{Iy} = \frac{\left(\rho_y F_x + \pi_y F_y + \eta_y F_z\right)}{m} \qquad [2.41\text{-}5]$$

$$a_{Iz} = \frac{\left(\rho_z F_x + \pi_z F_y + \eta_z F_z\right)}{m} - g \qquad [2.41\text{-}6]$$

where: m = mass
g = acceleration due to gravity

Under certain circumstances (see design element 41-4), a secondary set of Euler angles is defined thereby changing the terms in the direction cosine transformation matrix. The inertial to body secondary transformation matrix is:

$$\begin{bmatrix} a_{Ix} \\ a_{Iy} \\ a_{Iz} \end{bmatrix} = \begin{bmatrix} \cos\psi\cos\varphi - \sin\psi\cos\theta\sin\varphi & -\cos\psi\sin\varphi - \sin\varphi\cos\theta\cos\varphi & \sin\psi\sin\theta \\ \sin\psi\cos\varphi + \cos\psi\cos\theta\sin\varphi & -\sin\psi\sin\varphi + \cos\psi\cos\theta\cos\varphi & -\cos\psi\sin\theta \\ \sin\theta\sin\varphi & \sin\theta\cos\varphi & \cos\theta \end{bmatrix} \begin{bmatrix} F_x / m \\ F_y / m \\ F_z / m \end{bmatrix} \qquad [2.41\text{-}7]$$

When this matrix is in use, the terms ρ_x through η_z in equations 2.41-4, 5, and 6 are changed accordingly.

Design Element 41-2: Compute Body Angular Accelerations

In the rotational form of Newton's equation:

$$M = \dot{\omega} I \qquad [2.41\text{-}8]$$

where: M = Moment
$\dot{\omega}$ = Angular acceleration
I = Moment of Inertia

Solving for $\dot{\omega}$:

$$\dot{\omega} = \frac{M}{I} \qquad [2.41\text{-}9]$$

Resolving into body axis components and using defined terms for roll, pitch, and yaw rates gives:

$$\dot{p} = \frac{M_x}{A} \qquad [2.41\text{-}10]$$

$$\dot{q} = \frac{M_y}{B} \qquad [2.41\text{-}11]$$

$$\dot{r} = \frac{M_z}{C} \qquad [2.41\text{-}12]$$

For an aerodynamic vehicle such as a missile, it is not unreasonable to assume that the moment of inertia about the y-body axis is equal to that about the z-body axis which makes $B = C$. Another assumption which is made by ESAMS (due to lack of information) is to assume the roll rate is held to 0. After these assumptions, Equations 2.41-10, -11, and -12 become:

$$\dot{p} = 0 \qquad [2.41\text{-}13]$$

$$\dot{q} = \frac{M_y}{I} \qquad [2.41\text{-}14]$$

$$\dot{r} = \frac{M_z}{I} \qquad [2.41\text{-}15]$$

where: I = Moment of inertia around the y- and/or z-body direction.

Design Element 41-3: Calculation of 1st and 2nd Derivatives of Primary Euler Angles

Before the Euler angles can be integrated over time, their first and second derivatives must be derived. Reference 26 page 91 gives the angular rates in terms of the Euler angles as:

$$x-\text{axis},\quad p = \dot{\varphi} - \dot{\psi}\,sin\,\theta \tag{2.41-16}$$

$$y-\text{axis},\quad q = \dot{\theta}\,cos\,\varphi + \dot{\psi}\,sin\,\varphi\,cos\,\theta \tag{2.41-17}$$

$$z-\text{axis},\quad r = \dot{\psi}\,cos\,\varphi\,cos\,\theta - \dot{\theta}\,sin\,\varphi \tag{2.41-18}$$

where: θ = Pitch angle, positive up
Ψ = Yaw or heading angle, positive right
φ = Roll angle, positive clockwise from rear

In ESAMS, however, ψ and *r* are positive to the left. If we change the signs of these terms to make them correct for ESAMS we get:

$$x-\text{axis},\quad p = \dot{\varphi} + \dot{\psi}\,sin\,\theta \tag{2.41-19}$$

$$y-\text{axis},\quad q = \dot{\theta}\,cos\,\varphi - \dot{\psi}\,sin\,\varphi\,cos\,\theta \tag{2.41-20}$$

$$z-\text{axis},\quad -r = \dot{\psi}\,cos\,\varphi\,cos\,\theta - \dot{\theta}\,sin\,\varphi \tag{2.41-21}$$

Multiplying through the negative sign in equation 2.41-21 gives:

$$r = \dot{\psi}\,cos\,\varphi\,cos\,\theta + \dot{\theta}\,sin\,\varphi \tag{2.41-22}$$

Solving for angular rates of the Euler angles gives:

$$\dot{\psi} = \frac{(r\,cos\,\varphi - q\,sin\,\varphi)}{cos\,\theta} \tag{2.41-23}$$

$$\dot{\theta} = q\,cos\,\varphi + r\,sin\,\varphi \tag{2.41-24}$$

$$\dot{\varphi} = p + \frac{(q\,sin\,\varphi - r\,cos\,\varphi)\,sin\,\theta}{cos\,\theta} \tag{2.41-25}$$

These equations are the first derivatives of the primary Euler angles and represent their rates or velocities. Taking the derivatives of these, we get:

$$\ddot{\psi} = \frac{\dot{r}\,cos\,\varphi - \dot{q}\,sin\,\varphi + \dot{\theta}(\dot{\psi}\,sin\,\theta - \dot{\varphi})}{cos\,\theta} \tag{2.41-26}$$

$$\ddot{\theta} = \dot{q}\cos\varphi + \dot{r}\sin\varphi + \dot{\varphi}\dot{\psi}\cos\theta \qquad [2.41\text{-}27]$$

$$\ddot{\varphi} = \dot{p} + \frac{(\dot{q}\sin\varphi - \dot{r}\cos\varphi)\sin\theta}{\cos\theta} + \frac{(\dot{\varphi}\sin\theta - \dot{\psi})\dot{\theta}}{\cos\theta} \qquad [2.41\text{-}28]$$

These equations are the second derivatives of the primary Euler angles and represent their rates of change or accelerations. These two sets of equations can now be used in the integration of primary Euler angles over time.

Design Element 41-4: Calculation of 1st and 2nd Derivatives of Secondary Euler Angles

Inspection of equations 2.41-16 and 2.41-18 reveals a singularity when cosθ=0, or θ=90°. One method which can be employed to avoid this problem is to define a secondary set of Euler angles to be used when θ approaches 90° to which rotations will be applied. From reference 2, page 127, the secondary set of Euler angles defined by ESAMS are as follows:

ψ_a = rotation about the inertial Z-axis
θ_a = rotation about the intermediate X'-axis
φ_a = rotation about the z-body axis

This changes the order of rotations from yaw-pitch-roll to yaw-roll-yaw and also redefines θ as the roll angle and φ as an additional yaw angle. Figure 2.41-3 shows these angular relations and the secondary angular rotations. This combination of rotations eliminates the division by *cos*θ in the calculation of first and second derivatives. (As currently coded, ESAMS was designed to use alternate Euler angles as described here. The implementation has had some difficulties that have required suspending the use of alternate Euler angles. ESAMS 2.6.2 street MDR#1, dated 3 May 1995, acknowledges this problem.)

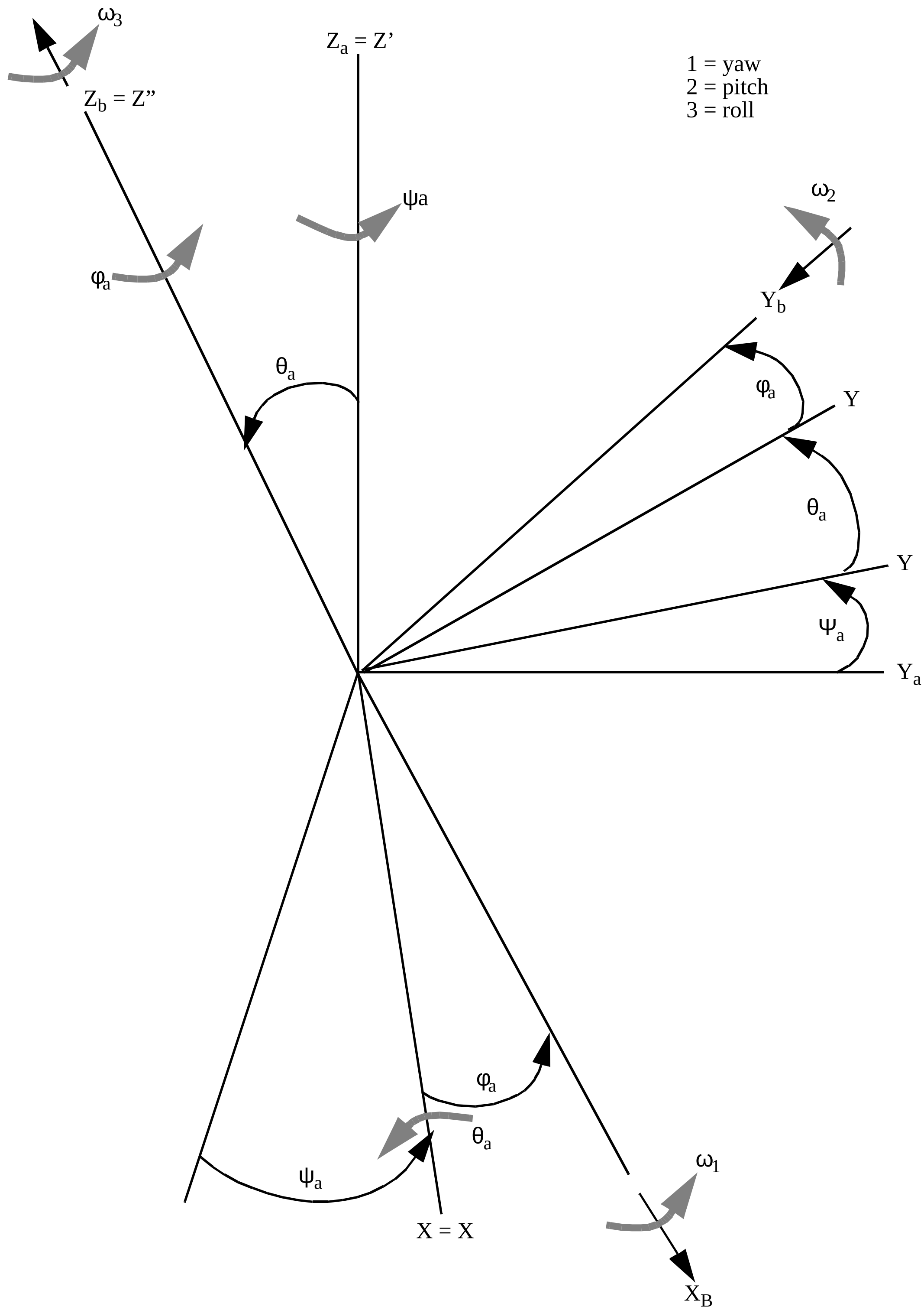

FIGURE 2.41-3. Alternate Missile Euler Angles.

Using these relationships, Reference 2 gives the angular rates in terms of alternate Euler angles using ESAMS sign conventions as:

$$p = \psi_a \sin\theta_a \sin\varphi_a + \dot{\theta}_a \cos\varphi_a \qquad [2.41\text{-}29]$$

$$q = \dot{\theta}_a \sin\varphi_a - \dot{\psi}_a \sin\theta_a \cos\varphi_a \qquad [2.41\text{-}30]$$

$$r = \dot{\varphi}_a + \dot{\psi}_a \cos\theta_a \qquad [2.41\text{-}31]$$

Solving for angular rates of secondary Euler angles gives:

$$\dot{\psi}_a = \frac{\left(p \sin\varphi_a - q \cos\varphi_a\right)}{\sin\theta_a} \qquad [2.41\text{-}32]$$

$$\dot{\theta}_a = p\cos\varphi_a + q\sin\varphi_a \qquad [2.41\text{-}33]$$

$$\dot{\varphi}_a = \frac{\left(q\cos\varphi_a - p\sin\varphi_a\right)\cos\theta_a}{\sin\theta_a} + r \qquad [2.41\text{-}34]$$

These equations are the first derivatives of the secondary Euler angles. Taking the derivatives of these, we get:

$$\ddot{\psi}_a = \frac{\left(\dot{p}\sin\varphi_a - \dot{q}\cos\varphi_a\right)}{\sin\theta_a} + \frac{\left(\dot{\varphi}_a - \psi_a \cos\theta_a\right)\dot{\theta}_a}{\sin\theta_a} \qquad [2.41\text{-}35]$$

$$\ddot{\theta}_a = \dot{p}\cos\varphi_a + \dot{q}\sin\varphi_a - \dot{\psi}_a \dot{\varphi}_a \sin\theta_a \qquad [2.41\text{-}36]$$

$$\ddot{\varphi}_a = \frac{\left(\dot{q}\cos\varphi_a - \dot{p}\sin\varphi_a\right)\cos\theta_a}{\sin\theta_a} + \dot{r} + \frac{\left(\psi_a - \dot{\varphi}_a \cos\theta_a\right)\dot{\theta}_a}{\sin\theta_a} \qquad [2.41\text{-}37]$$

These equations are the second derivatives of the secondary Euler angles. These two sets of equations can now be used in the integration of secondary Euler angles over time. It should be restated that these sets of secondary Euler angles are only used if θ approaches 90°.

Design Element 41-5: Convert Euler Angles to Alternate Set

When a switch from one set of Euler angles to the other occurs, the "old" angles must be converted to the "new" angles as defined in the alternate reference system. Values for the "old" angles are preserved in the direction cosine transformation matrix terms. In this way, the two systems are linked to each other by the transformation matrix. By definition, the appropriate terms from each matrix are equal to each other. That is, ρ_x from the primary set is equal to ρ_x from the secondary set. Therefore, like terms can be set equal and solved for either primary or secondary Euler angles. This can be done with any combination of the applicable terms with some being much easier to use than others.

After careful examination of the secondary matrix terms, the most convenient relationships to relate to primary angles are chosen as follows.

To find θ_a from known primary terms,

$$\eta_z = cos\theta_a \quad [2.41\text{-}38]$$

Solving for θ_a,

$$\theta_a = cos^{-1}[\eta_z]^{primary} \quad [2.41\text{-}39]$$

To find ψa from known primary terms, we need two equations with two unknowns,

$$\eta_x = sin\psi_a sin\theta_a$$

$$-\eta_x = cos\psi_a sin\theta_a \quad [2.41\text{-}40a,b]$$

Dividing these equations and solving for ψ_a:

$$\psi_a = tan^{-1}\left[\frac{\eta_x}{-\eta_y}\right]^{primary} \quad [2.41\text{-}41]$$

To find φ_a from known primary terms, we need two equations with two unknowns,

$$\rho_z = sin\theta_a sin\varphi_a$$

$$\pi_z = sin\theta_a cos\varphi_a \quad [2.41\text{-}42a,b]$$

Dividing these equations and solving for φ_a,

$$\varphi_a = tan^{-1}\left[\frac{\rho_z}{\pi_z}\right]^{primary} \quad [2.41\text{-}43]$$

After careful examination of the primary matrix terms, the most convenient relationships to relate to secondary angles are chosen as follows.

To find θ from known secondary terms,

$$\rho_z = sin\theta \quad [2.41\text{-}44]$$

Solving for θ,

$$\theta = sin^{-1}[\rho_z]^{secondary} \quad [2.41\text{-}45]$$

To find ψ from known secondary terms, we need two equations with two unknowns,

$$\rho_y = sin\psi cos\theta$$

$$\rho_x = cos\psi cos\theta \quad [2.41\text{-}46a,b]$$

Dividing these equations and solving for ψ,

$$\psi = tan^{-1}\left[\frac{\rho_y}{\rho_x}\right]^{secondary} \quad [2.41\text{-}47]$$

To find φ from known secondary terms we need two equations with two unknowns,

$$\pi_z = cos\theta sin\varphi$$
$$\eta_z = cos\theta cos\varphi \quad [2.41\text{-}48a,b]$$

Dividing these equations and solving for φ,

$$\varphi = tan^{-1}\left[\frac{\pi_z}{\eta_z}\right]^{secondary} \quad [2.41\text{-}49]$$

Design Element 41-6: Integrate Inertial Positions and Euler Angles

If acceleration is assumed to be constant over the time interval (Δt), the equation for the new position of the body (either linear or angular) is given by:

$$S(t+\Delta t) = S_0 + V_0(t)\Delta t + \frac{1}{2}a\Delta t^2 \quad [2.41\text{-}50]$$

where: S_0 = position at end of last time step
V_0 = velocity at end of last time step
a = current acceleration

Even if the acceleration is changing, this is a good approximation for small Δt. The position equation can be used to update the missile's x, y and z inertial location as well as the Euler angles which represent the missile's orientation over the time step. Using appropriate terms for the inertial location:

$$X = X_0 + V_{0I_x}\Delta t + \frac{1}{2}\Delta t^2 a_{I_x} \quad [2.41\text{-}51]$$

$$Y = Y_0 + V_{0I_y}\Delta t + \frac{1}{2}\Delta t^2 a_{I_y} \quad [2.41\text{-}52]$$

$$Z = Z_0 + V_{0I_z}\Delta t + \frac{1}{2}\Delta t^2 a_{I_z} \quad [2.41\text{-}53]$$

where: $V_{0Ix,y,z}$ = Inertial components of velocity at end of last time step
$a_{Ix,y,z}$ = Inertial components of acceleration

Using the appropriate terms for Euler angles:

$$\psi = \psi_0 + \dot{\psi}_0\Delta t + \frac{1}{2}\ddot{\psi}\Delta t^2 \quad [2.41\text{-}54]$$

$$\theta = \theta_0 + \dot{\theta}_0\Delta t + \frac{1}{2}\ddot{\theta}\Delta t^2 \quad [2.41\text{-}55]$$

$$\varphi = \varphi_0 + \dot{\varphi}_0 \Delta t + \frac{1}{2}\ddot{\varphi}\Delta t^2 \qquad [2.41\text{-}56]$$

where: ψ_0, θ_0, and φ_0 = Euler angles and rates at end of last time step

When an appropriate time step is chosen, these two sets of equations can be used to update the missile's inertial position and orientation.

Design Element 41-7: Integrate Inertial Velocities and Angular Rates

If acceleration is assumed to be constant over the time interval (Δt), the equation for the new velocity of the body (either linear or angular) is given by:

$$V(t + \Delta t) = V_0(t) + a\Delta t \qquad [2.41\text{-}57]$$

where: V_0 = velocity at end of last time step
a = current acceleration

Even if the acceleration is changing, this is a good approximation for small Δt. The velocity equation can be used to update the missile's x, y and z inertial velocities as well as the angular rates over the time step. Using appropriate terms for the inertial velocities:

$$V_{I_x} = V_{0I_x} + a_{I_x}\Delta t \qquad [2.41\text{-}58]$$

$$V_{I_y} = V_{0I_y} + a_{I_y}\Delta t \qquad [2.41\text{-}59]$$

$$V_{I_z} = V_{0I_z} + a_{I_z}\Delta t \qquad [2.42\text{-}60]$$

where: $V_{0Ix,y,z}$ are inertial velocities at the end of the last time step

Using the appropriate terms for angular rates:

$$r = r_0 + \dot{r}\Delta t \qquad [2.41\text{-}61]$$

$$q = q_0 + \dot{q}\Delta t \qquad [2.41\text{-}62]$$

$$p = p_0 + \dot{p}\Delta t \qquad [2.41\text{-}63]$$

When an appropriate time step is chosen, these two sets of equations can be used to update the missile's inertial velocity and angular rates

2.41.3 Functional Element Software Design

This section contains the software design necessary to implement the functional element requirements described in Section 2.41.1 and the design approach described in Section 2.41.2. Section 2.41.3 is organized as follows: the first section describes the subroutine hierarchy and gives descriptions of the relevant subroutines; the next subsection contains logical flow charts and describes all important operations represented by each block in the charts; the last subsection contains a description of all input and output data for the functional element as a whole and for each subroutine that implements force and moment generation.

Missile Movement Subroutine Design

The FORTRAN call tree implemented for the Missile Movement Functional Element in ESAMS code is shown in Figure 2.41-4. The diagram depicts the structure of the entire model for this functional element, from ZINGER (the main program) through the least significant subroutine implementing missile movement. Subroutines which directly implement the functional element appear as shaded blocks. Each of these subroutines is described briefly in Table 2.41-2.

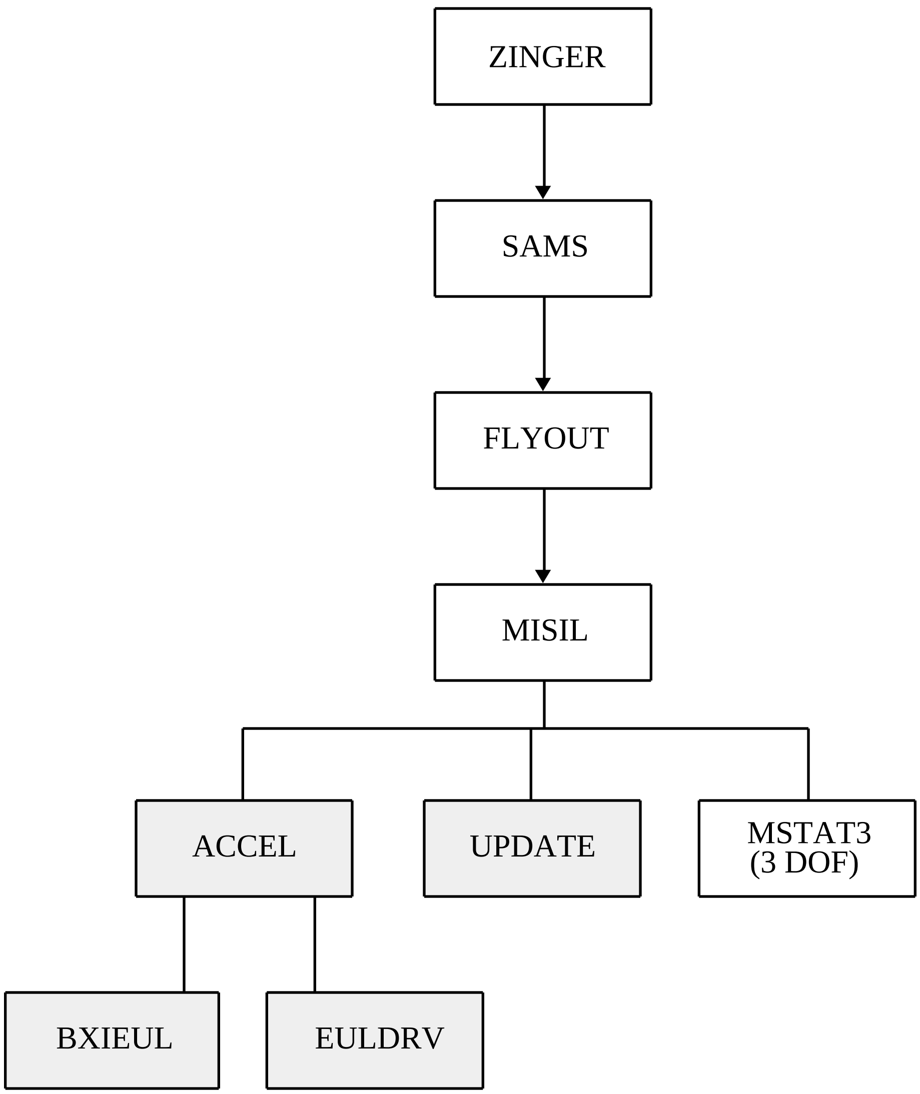

FIGURE 2.41-4. Call Hierarchy for Missile Movement.

TABLE 2.41-2. Subroutine Descriptions.

Module Name	Description
ACCEL	Computes body and inertial linear accelerations and body angular accelerations. Determines if switch from primary to secondary or secondary to primary Euler angles is required.
BXIEUL	Computes either primary or secondary Euler angles from the current attitude matrix if Euler angles switch in either direction.
EULDRV	Calculates first and second derivatives of either set of Euler angles
UPDATE	Initializes current position, velocity, Euler angles and angular rates. Integrates position, Euler angles, velocity and angular rates, increments integration step counter and autopilot time.
MSTAT3	Calculate forces and moments and updates position and velocity for 3 DOF model.
Note: Modules implementing the missile movement functional element for 5 DOF models are identified in bold letters	

Functional Flow Diagram

Figure 2.41.5 shows the top-level logical flow of missile movement implementation. Subroutine names appear in the parentheses at the bottom of each process block. The numbered blocks are described below.

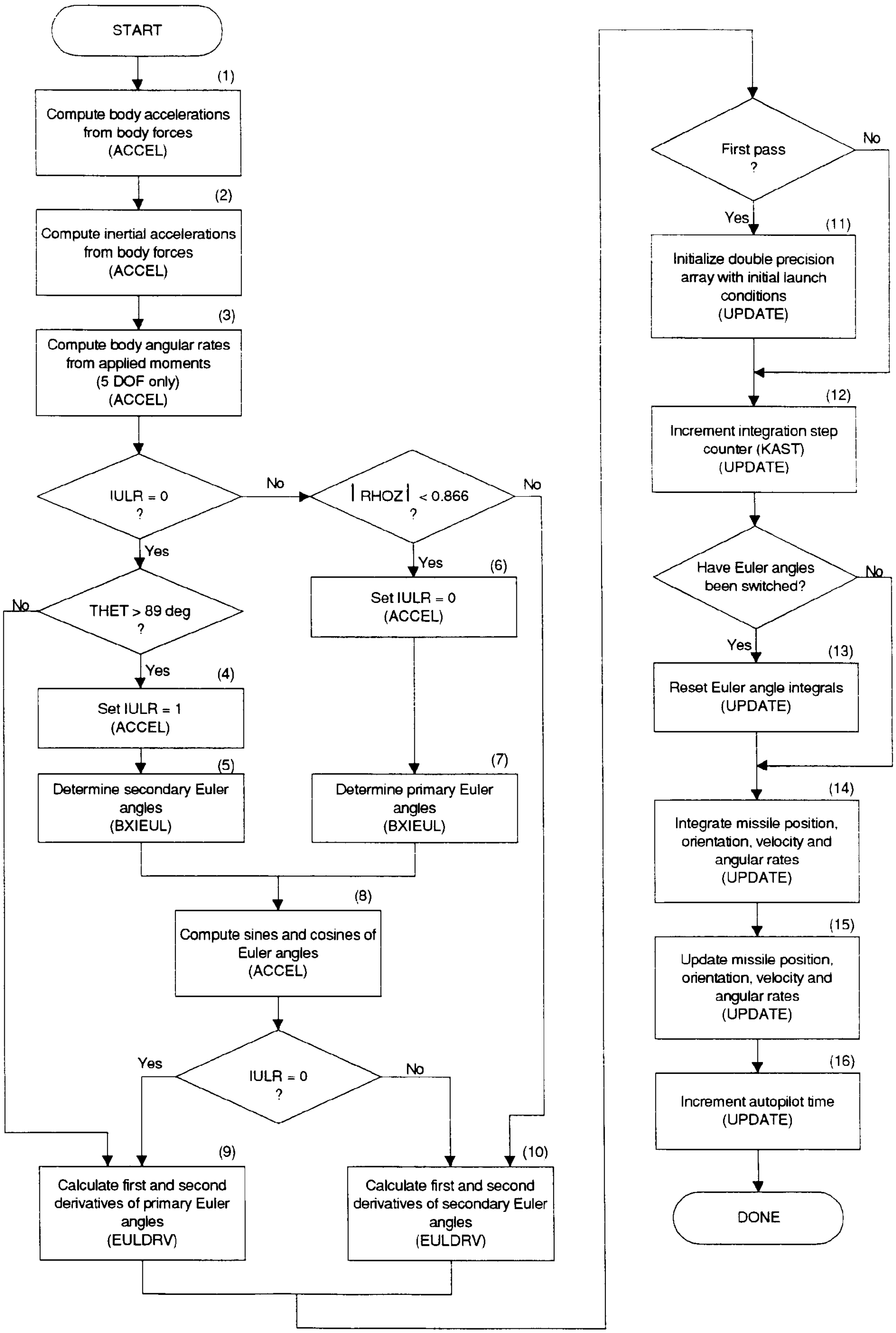

FIGURE 2.41-5. Missile Movement Logical Flow.

Block 1. The components of body accelerations are computed from body forces, F_x, F_y, and F_z.

Block 2. The components of inertial accelerations are computed from body forces and gravity using Equations [2.41-4,-5 and -6].

Block 3. The components of body angular accelerations are computed from applied moments and moment of inertial using Equations [2.41-13, -14 and -15] for 5 DOF cases only. Moments and angular accelerations are not computed for 3 DOF models.

Block 4. If primary Euler angles are in use and missile pitch angle (THET) is greater than 89°, the Euler angle switching flag, IULR, is set to 1.

Block 5. With IULR=1, the secondary Euler angles are computed from terms in the primary direction cosine transformation matrix using Equations [2.41-39, -41 and -43].

Block 6. If secondary Euler angles are in use, IULR=1, and absolute value of ρ_z is less than 0.866 then missile pitch has changed enough to switch the Euler angle flag, IULR, to 0.

Block 7. With IULR=0, the primary Euler angles are computed from terms in the secondary direction cosine transformation matrix using Equations [2.41-45, -47 and -49].

Block 8. If Euler angles have switched and after they have been recomputed within BXIEUL, the sines and cosines of the Euler angles are computed.

Block 9. If primary Euler angles are in use, IULR=0, and missile pitch angle (THET) is less than 89˚, the first derivatives of the primary Euler angles are computed using Equations [2.41-23, -24, and -25] and the second derivatives of the primary Euler angles are computed using Equations [2.41-26, -17, and -28].

Block 10. If secondary Euler angles are in use, the first derivatives of the secondary Euler angles are computed using Equations [2.41-32, -33 and -34] and the second derivatives of the secondary Euler angles are computed using Equations [2.41-35, -36 and -37].

Block 11. On the first pass through the model, the double precision array values for position, velocity, Euler angles and angular rates are set to initial values based on launch conditions.

Block 12. The missile integration time step, KAST, is incremented by one on every pass.

Block 13. If the Euler angles have switched, the double precision Euler angle integrals (representing the current Euler angles to be updated, θ_0, φ_0, and ψ_0 in Equations [2.41-54, -55 and 56]) need to be reset to the current (primary or secondary) values. If this is not done, the delta value added to update an Euler angle will be of the alternate set of angles.

Block 14. The missile's position is integrated using Equations [2.41-51, -52 and 53], velocity is integrated using Equations [2.41-58, -59 and -60], Euler angles are integrated using Equations [2.41-54, -55 and -56] and angular rates are integrated using Equations [2.41-61, -62 and 63].

Block 15. The variables representing missile position, velocity, Euler angles and angular rates are updated to the new values.

Block 16. The autopilot time, ATIME, is incremented by the autopilot integration step.

Missile Movement Inputs and Outputs

The outputs of this functional element are position, orientation, velocity and acceleration information given in Table 2.41-3. User inputs which affect missile movement are given in Table 2.41-4.

TABLE 2.41-3. Missile Movement Outputs.

Variable Name	Description
XDDB	Missile acceleration in the x-body direction (m/sec2)
YDDB	Missile acceleration in the y-body direction (m/sec2)
ZDDB	Missile acceleration in the z-body direction (m/sec2)
XDDOT	Missile inertial acceleration in the X direction, aIx (m/sec2)
YDDOT	Missile inertial acceleration in the Y direction, aIy (m/sec2)
ZDDOT	Missile inertial acceleration in the Z direction, aIz (m/sec2)
OMEGD(1)	Angular acceleration around the x-body axis, p (rad/sec2)
OMEGD(2)	Angular acceleration around the y-body axis, q (rad/sec2)
OMEGD(3)	Angular acceleration around the z-body axis, r (rad/sec2)
IULR	Euler angle switching flag, 0=primary, 1=secondary
THET	Missile Euler angle for pitch, q (rad)
PSI	Missile Euler angle for yaw, y (rad)
PHI	Missile Euler angle for roll, f (rad)
X	Missile inertial X position (m)
Y	Missile inertial Y position (m)
Z	Missile inertial Z position (m)
XDOT	Missile inertial velocity in X direction (m/sec)
YDOT	Missile inertial velocity in Y direction (m/sec)
ZDOT	Missile inertial velocity in Z direction (m/sec)
OMEG(1)	Rotation rate around x-body direction, p (rad/sec)
OMEG(2)	Rotation rate around y-body direction, q (rad/sec)
OMEG(3)	Rotation rate around z-body direction, r (rad/sec)
CTHT	Cosine of THET
STHT	Sine of THET
SPSI	Sine of PSI
CPSI	Cosine of PSI
CPHI	Cosine of PHI
SPHI	Sine of PHI
KAST	Missile integration step counter
ATIME	Missile autopilot time

TABLE 2.41-4. User Inputs for Missile Movement.

Common Name	Variable Name	Description
ROPTN	IXCRD	Missile axes configuration selection, 0=0°, 1=45°
GUIDAP	DTA	Autopilot integration step

Inputs and outputs for the major routines implementing missile movement functional element are given in Tables 2.41-5 through 2.41-8. The inputs and outputs related to missile movement and for 5-DOF simulations only are printed in bold.

TABLE 2.41-5. Subroutine ACCEL Inputs and Outputs.

SUBROUTINE: ACCEL					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
THRUST	Common MISSIL	Missile engine thrust (N)	XDDB	Common MISSIL	Missile acceleration in x-body direction (m/sec2)
FA(1)	Common MISSIL	Missile x-body aero force vector (N)	YDDB	Common MISSIL	Missile acceleration in y-body direction (m/sec2)
FA(2)	Common MISSIL	Missile y-body aero force vector (N)	ZDDB	Common MISSIL	Missile acceleration in z-body direction (m/sec2)
FA(3)	Common MISSIL	Missile z-body aero force vector (N)	XDDOT	Common MISSIL	Missile inertial acceleration in X direction (m/sec2)
FMASS	Common MISSIL	Missile mass	YDDOT	Common MISSIL	Missile inertial acceleration in Y direction (m/sec2)
G	Common PARAM	Acceleration due to gravity (m/sec2)	ZDDOT	Common MISSIL	Missile inertial acceleration in Z direction (m/sec2)
AINERT	Common MISSIL	Missile moment of inertia (kg m2)	OMEGD(1)	Common MISSIL	Angular acceleration around x-body axis (rad/sec2)
RHOX	Common EULER	Term from direction cosine transformation matrix relating missile body x-direction to inertial X-direction	OMEGD(2)	Common MISSIL	Angular acceleration around y-body axis (rad/sec2)
RHOY	Common EULER	Term from direction cosine transformation matrix relating missile body x-direction to inertial Y-direction	OMEGD(3)	Common MISSIL	Angular acceleration around z-body axis (rad/sec2)
RHOZ	Common EULER	Term from direction cosine transformation matrix relating missile body x-direction to inertial Z-direction	GEE	Common MISSIL	Missile G force

TABLE 2.41-5. Subroutine ACCEL Inputs and Outputs. (Contd.)

SUBROUTINE: ACCEL					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
PIX	Common EULER	Term from direction cosine transformation matrix relating missile body y-direction to inertial X-direction	IULR	Common SIMVI	Euler angle selection flag
PIY	Common EULER	Term from direction cosine transformation matrix relating missile body y-direction to inertial Y-direction	SPSI	Common EULER	Sine of PSI
PIZ	Common EULER	Term from direction cosine transformation matrix relating missile body y-direction to inertial Z-direction	CPSI	Common EULER	Cosine of PSI
ETAX	Common EULER	Term from direction cosine transformation matrix relating missile body z-direction to inertial X-direction	STHT	Common EULER	Sine of THET
ETAY	Common EULER	Term from direction cosine transformation matrix relating missile body z-direction to inertial Y-direction	CTHT	Common EULER	Cosine of THET
ETAZ	Common EULER	Term from direction cosine transformation matrix relating missile body z-direction to inertial Z-direction	SPHI	Common EULER	
TAUA(2)	Common MISSIL	Moment around y-body axis (Nm)	CPHI	Common EULER	Cosine of PHI
TAUA(3)	Common MISSIL	Moment around z-body axis (Nm)			
IAUTO	Common ROPTN	Number of times to go through autopilot			
IULR	Common SIMVI	Primary or secondary Euler angle selection			
PSI	Common EULER	Missile yaw angle (deg)			
PHI	Common EULER	Missile roll angle (deg)			
THET	Common EULER	Missile pitch angle (deg)			

TABLE 2.41-6. Subroutine BXIEUL Inputs and Outputs.

SUBROUTINE: BXIEUL					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
RHOX	Common EULER	Term from direction cosine transformation matrix relating missile body x-direction to inertial X-direction	PSI	Common EULER	Missile yaw angle (deg)
RHOY	Common EULER	Term from direction cosine transformation matrix relating missile body x-direction to inertial Y-direction	THET	Common EULER	Missile pitch angle (deg)
RHOZ	Common EULER	Term from direction cosine transformation matrix relating missile body x-direction to inertial Z-direction	PHI	Common EULER	Missile roll angle (deg)
PIZ	Common EULER	Term from direction cosine transformation matrix relating missile body y-direction to inertial Z-direction			
ETAX	Common EULER	Term from direction cosine transformation matrix relating missile body z-direction to inertial X-direction			
ETAY	Common EULER	Term from direction cosine transformation matrix relating missile body z-direction to inertial Y-direction			
ETAZ	Common EULER	Term from direction cosine transformation matrix relating missile body z-direction to inertial Z-direction			
IUL	Argument	Primary or secondary Euler angle selection			
IXCRD	Common ROPTN	Missile axes configuration selection			
PIO2	Common PARAM	Pi over 2			
PIO4	Common PARAM	Pi over 4			

TABLE 2.41-7. Subroutine EULDRV Inputs and Outputs.

SUBROUTINE: EULDRV					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
SPSI	Common EULER	Sine of PSI	PSID	Common EULER	First derivative of PSI
CPSI	Common EULER	Cosine of PSI	THETD	Common EULER	First derivative of THET
STHT	Common EULER	Sine of THET	PHID	Common EULER	First derivative of PHI
CTHT	Common EULER	Cosine of THET	PSIDD	Common EULER	Second derivative of PSI
SPHI	Common EULER	Sine of PHI	THETDD	Common EULER	Second derivative of THET
CPHI	Common EULER	Cosine of PHI	PHIDD	Common EULER	Second derivative of PHI
OMEG(1)	Common MISSIL	Angular velocity around x-body axis (rad/sec)			
OMEG(2)	Common MISSIL	Angular velocity around y-body axis (rad/sec)			
OMEG(3)	Common MISSIL	Angular velocity around z-body axis (rad/sec)			
OMEGD(1)	Common MISSIL	Angular acceleration around x-body axis (rad/sec2)			
OMEGD(2)	Common MISSIL	Angular acceleration around y-body axis (rad/sec2)			
OMEGD(3)	Common MISSIL	Angular acceleration around z-body axis (rad/sec2)			
DTA	Common GUIDAP	Autopilot integration step			
IULR	Common SIMVI	Primary or secondary Euler angle selection			

TABLE 2.41-8. Subroutine UPDATE Inputs and Outputs.

SUBROUTINE: UPDATE					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
KAST	Common SIMVI	Missile integration step counter	KAST	Common MSIMVI	Missile integration step counter
X	Common MISSIL	Initial missile inertial X position (m)	X	Common MISSIL	Initial missile inertial X position (m)
Y	Common MISSIL	Initial missile inertial Y position (m)	Y	Common MISSIL	Initial missile inertial Y position (m)

TABLE 2.41-8. Subroutine UPDATE Inputs and Outputs. (Contd.)

SUBROUTINE: UPDATE					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
Z	Common MISSIL	Initial missile inertial Z position (m)	Z	Common MISSIL	Initial missile inertial Z position (m)
XDOT	Common MISSIL	Missile inertial velocity in X direction (m/sec)	XDOT	Common MISSIL	Missile inertial velocity in X direction (m/sec)
YDOT	Common MISSIL	Missile inertial velocity in Y direction (m/sec)	YDOT	Common MISSIL	Missile inertial velocity in Y direction (m/sec)
ZDOT	Common MISSIL	Missile inertial velocity in Z direction (m/sec)	ZDOT	Common MISSIL	Missile inertial velocity in Z direction (m/sec)
PSI	Common EULER	Missile yaw angle (rad)	PSI	Common EULER	Missile yaw angle (rad)
THET	Common EULER	Missile pitch angle (rad)	THET	Common EULER	Missile pitch angle (rad)
PHI	Common EULER	Missile roll angle (rad)	PHI	Common EULER	Missile roll angle (rad)
IULR	Common SIMVI	Primary or secondary Euler angle selection	OMEG(1)	Common MISSIL	Angular velocity around x-body axis (rad/sec)
XDDOT	Common MISSIL	Missile inertial acceleration in X direction (m/sec2)	OMEG(2)	Common MISSIL	Angular velocity around y-body axis (rad/sec)
YDDOT	Common MISSIL	Missile inertial acceleration in Y direction (m/sec2)	OMEG(3)	Common MISSIL	Angular velocity around z-body axis (rad/sec)
ZDDOT	Common MISSIL	Missile inertial acceleration in Z direction (m/sec2)	ATIME	Common GUIDAP	Autopilot time
DTA	Common GUIDAP	Autopilot integration step			
PSID	Common EULER	First derivative of PSI			
THETD	Common EULER	First derivative of THET			
PHID	Common EULER	First derivative of PHI			
PSIDD	Common EULER	Second derivative of PSI			
THETDD	Common EULER	Second derivative of THET			
PHIDD	Common EULER	Second derivative of PHI			
OMEGD(1)	Common MISSIL	Angular acceleration around x-body axis (rad/sec2)			
OMEGD(2)	Common MISSIL	Angular acceleration around y-body axis (rad/sec2)			

TABLE 2.41-8. Subroutine UPDATE Inputs and Outputs. (Contd.)

SUBROUTINE: UPDATE					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
OMEGD(3)	Common MISSIL	Angular acceleration around z-body axis (rad/sec2)			
DXI(1)	Common INTEG	Value of X from last time step			
DXI(2)	Common INTEG	Value of Y from last time step			
DXI(3)	Common INTEG	Value of Z from last time step			
DXI(4)	Common INTEG	Value of XDOT from last time step			
DXI(5)	Common INTEG	Value of YDOT from last time step			
DXI(6)	Common INTEG	Value of ZDOT from last time step			
DXI(7)	Common INTEG	Value of PSI from last time step			
DXI(8)	Common INTEG	Value of THET from last time step			
DXI(9)	Common INTEG	Value of PHI from last time step			
DXI(10)	Common INTEG	Value of OMEG(1) from last time step			
DXI(11)	Common INTEG	Value of OMEG(2) from last time step			
DXI(12)	Common INTEG	Value of OMEG(3) from last time step			
ATIME	Common GUIDAP	Autopilot time			

2.41.4 Assumptions and Limitations

The missile is assumed to be perfectly roll stabilized, that is, roll rate is held to 0.

The missile is assumed to have identical symmetry about the y and z body axes.

Acceleration and missile physical properties (thrust, mass, CG, etc.) are assumed to be constant over the time step.